Krzysztof KONOPKO [1]

Bialystok University of Technology, Electrical Engineering Faculty, Białystok, Poland,

An implementation of the Cohen's class time-frequency distributions on a massively parallel processor

***Abstract**. A time-frequency signal analysis (TFSA) refers to processing of signals with a time-varying frequency content. These non-stationary signals are best represented by the time-frequency distributions, which show how the energy of the signal is distributed over a two-dimensional time-frequency space. Real time processing of these non-stationary signals demands the computational performance of a few giga operations per second, which cannot be obtained using the single or multi core processors. The required computing power can be provided by the massively parallel processors. The paper presents an implementation of the Cohen's class time-frequency representations on a massively parallel processor.*

***Streszczenie.** Łączne czasowo-częstotliwościowe reprezentacje sygnałów umożliwiają analizowanie sygnałów, których widmo częstotliwościowe zmienia się w czasie. Przetwarzanie, w czasie rzeczywistym, sygnałów niestacjonarnych wymaga jednak znacznej mocy obliczeniowej. W artykule przedstawiono efektywną implementację czasowo-częstotliwościowej reprezentacji sygnałów z zastosowaniem procesorów masowo-równoległych. (**Realizacja czasowo-częstotliwościowej reprezentacji sygnałów z klasy Cohena z zastosowaniem procesorów masowo-równoległych**).*

Keywords: *a time-frequency signal analysis, massively parallel processors, opencl.*
Słowa kluczowe: *czasowo-częstotliwościowa analiza sygnałów, procesory masowo-równoległe, opencl.*

Introduction

Conventional spectral analysis of a signal has been based on Fourier transform techniques. However, this tool is useful only for analyzing signals whose characteristics do not vary with time. On the other hand, most of the signals emitted by natural or artificial systems had different forms of time dependence of their spectral content. This nonstationarity very often carry the most important information about a signal. This information can be extracted using a time-frequency representation (TFR), which is intended to show how the energy of the signal is distributed over the two-dimensional time-frequency space. The TFRs have been successfully used in numerous practical applications [2] in the areas of wireless communications, radar [3,4] and sonar systems, biomedical signal analyses and multimedia signal processing. However, these applications typically require large amounts of computing power, which cannot be obtained using the single or multi core processors. The required computing power can be delivered by modern graphics processing units (GPUs) which have evolved into massively parallel processor systems allowing very efficient manipulation of large blocks of data.

Time-Frequency Representations

One of the method for a nonstationary signal analysis is a decomposition of a signal into a set of blocks which can extract signal properties in time as well as in frequency. This decomposition for a signal $x(t)$ can be written as:

(1) $$x(t)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\lambda_x(t,f)h_{t,f}(t)dtdf\ ,$$

where function $h_{t,f}(t)$ plays a role of a time-frequency atom which possesses joint time-frequency localization properties. The inverse transform of (1) is given by:

(2) $$\lambda_x(t,f)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}x(t)h^{*}_{t,f}(t)dt\ ,$$

and $\lambda_x(t,f)$ can be interpreted as a linear time-frequency representation of $x(t)$. There is obviously a great arbitrariness in the choice of such a representation. For an example the choice of $h_{tf}(s)=h(s-t)\exp(j2\pi fs)$ leads to the family of short-time Fourier transforms with a window $h(.)$, and the choice of $h_{tf}(s)=\sqrt{f/f_0}h((f/f_0)s-t)$ leads to the family of wavelet transforms.

The numerous TFRs decompose not of the signal itself, but of its energy. The energy E_x of a signal $x(t)$ can be equivalently expressed as:

(3) $$E_x=\int_{-\infty}^{\infty}|x(t)|^2dt=\int_{-\infty}^{\infty}|X(f)|^2df\ .$$

The decomposition for the energy $x(t)$ at the time can be written as:

(4) $$E_x=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\rho_x(t,f)dtdf\ .$$

The choice of $\rho_x(t,f)$ can be carried out by various methods which results in a large number of different approaches. One of the most know is a spectrogram which can be defined as:

(5) $$S(t,f)=\left|\int_{-\infty}^{+\infty}x(t)h^{*}(t-\tau)e^{-j2\pi f\tau}dt\right|^2\ .$$

The length of window $h(.)$ determines the duration of the windowed signal $x(t)$. Improving the time localization by using a shorter window, results in a broadening of the spectrum and consequently the frequency localization is diminished. Of course, the reverse will happen if we lengthen the window. In that case the frequency localization is improved at the cost of the time localization. The effect of the window on the local spectrum will be minimal if the characteristics of the signal are not altered by the application of the window. The simplest way to this aim, is to use the signal itself as the window function:

(6) $$WV(t,f)=\int_{-\infty}^{+\infty}x(t+\frac{\tau}{2})x^{*}(t-\frac{\tau}{2})e^{-j2\pi f\tau}d\tau\ .$$

where $WV(t,f)$ is the Wigner-Ville distribution (WVD) [1] of the signal $x(t)$. The WVD exhibits the highest signal energy concentration in the time-frequency plane for linearly modulated signals, but has drawbacks in the cases of nonlinear frequency modulated signals. In addition to the WVD a number of bilinear distributions have been proposed [1, 2] for removing the cross terms. All of them can be expressed by the following equation:

$$(7)\quad S(t,f;\phi)=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\phi(\xi,\tau;t,f)\,x(s+\frac{\tau}{2})x^{*}(s-\frac{\tau}{2})e^{j2\pi[\xi(t-s)-f\tau]}ds\,d\tau\,d\xi\ .$$

This equation, known as Cohen’s class[1,2], allows to generate many useful representations by a specification of the kernel function $\phi(\xi,\tau;t,f)$.

For showing properties of the presented methods a nonstationary signal has been analysed. The signal consists of three components: a simple sinusoidal signal with a constant normalized frequency equal to 0.1 which lasts from 1st to 512th sample, a simple sinusoidal signal with a constant normalized frequency equal to 0.3 which lasts from 513th to 1024th sample and a chirp (linear swept-frequency) signal which starts at the first sample with normalized frequency equal 0.15 and ends at the last sample with normalized frequency equal 0.25.

The TFR of this three components signal is shown in Fig. 1-3. The Fig. 1 shows the spectrogram, the Fig. 2 presents the Wigner-Ville distribution and the Fig. 3 shows smooth pseudo Wigner-Ville distribution.

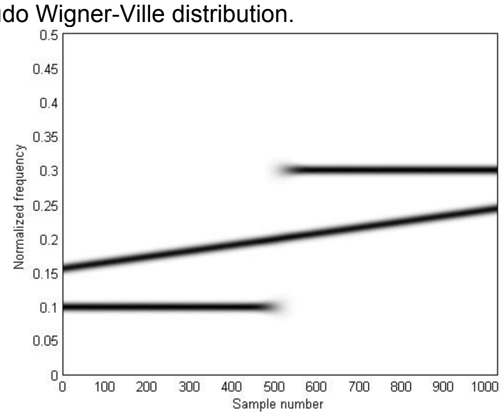

Figure. 1. The spectrogram of the three components signal with a sliding Blackman window of length N=128

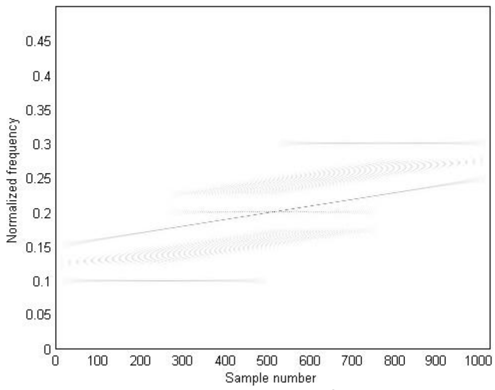

Figure. 2. The Wigner-Ville distribution of the three components signal.

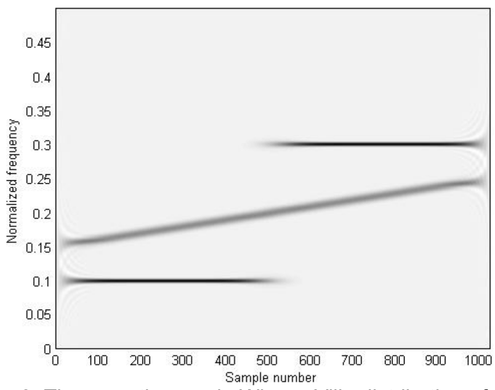

Figure. 3. The smooth-pseudo Wigner-Ville distribution of the three components signal.

GPUs - the massively parallel processors

The implementation of the Cohen’s class time-frequency representations requires large amounts of computing power, which can be delivered by modern graphics processing units (GPUs). General-Purpose computing using GPUs (GPGPU) has been an area of active research for a few years [6]. During this time, graphics processing units became massively parallel general-purpose processors. Nowadays GPUs have a very high floating point throughput (in excess of 5 TFlop/s for AMD Radeon HD 6990), a massive memory bandwidth (in excess of 320 GB/s for NVIDIA GeForce GTX590) and a design conducive to data parallelism. These features make them interesting in a nonstationary signal analysis, which demands these three properties. Moreover, to take advantage of the increased programmability provided by GPUs, languages and runtime systems such as CUDA [6], ATI Stream and OpenCL [5] have been developed. A common problem with the CUDA and ATI Stream frameworks is that they can only be used with NVIDIA’s or AMD’s GPUs. The OpenCL can be used not only with AMD’s or NVIDIA’s GPUs, but also with many different platforms and devices provided by a long range of developers.

OpenCL

The OpenCL framework can be divided into three essentials parts [5]: the platform part, which describes how OpenCL handles devices, the memory part which describes the different memory types on OpenCL devices and the execution part which describes how the parts of an application (memory objects and kernels) are executed by OpenCL. Execution of an OpenCL program occurs in two parts: kernels that execute on one or more OpenCL devices and a host program that executes on the host. The host program defines the context for the kernels and manages their execution. The kernels can be executed on a 1D, 2D, or 3D domain of indexes (NDRange) that execute in parallel, given enough resources. The total number of elements (indexes) in the launch domain is called the global work size; individual elements are known as work-items. Work-items can be grouped into work-groups when communication between work-items is required. Work-groups are defined with a sub-index function (called the local work size), describing the size in each dimension corresponding to the dimensions specified for the global launch domain. The Fig. 4 shows an example of an 3D domain index space.

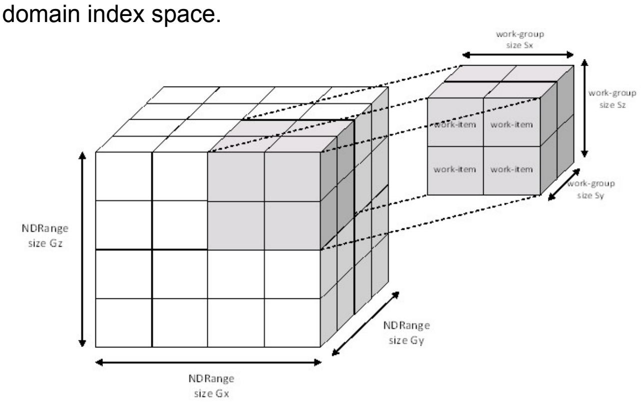

Figure. 4. An example of an 3D NDRange index space showing work-items grouped into a local work-group.

Cohen's class implementation on GPU

The implementation of Cohen's class can be carried out in three ways. Firstly as the Fourier transform of an instantaneous smooth autocorrelation function of the signal:

$$S(t,f)=\int_{-\infty}^{+\infty} r_x(t,\tau)e^{-j2\pi f\tau}d\tau, \tag{8}$$

where:

$$r_x(t,\tau)=\int_{-\infty}^{+\infty}\phi_{t-d}(t-s,\tau)x(s+\frac{\tau}{2})x^*(s-\frac{\tau}{2})ds. \tag{9}$$

Secondly as the result of the two-dimensional Fourier transform of a product of kernel function and the narrowband symmetric ambiguity function (11):

$$S_x(t,f)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\phi_{d-D}(\tau,\xi)A_x(\tau,\xi)e^{j2\pi(t\xi-f\tau)}d\xi d\tau, \tag{10}$$

where:

$$A_x(\tau,\xi)=\int_{-\infty}^{+\infty}x(t+\frac{\tau}{2})x^*(t-\frac{\tau}{2})e^{-j2\pi\xi\tau}dt. \tag{11}$$

And finally as the result of a two-dimensional convolution of the Wigner-Ville representation and the inverse Fourier transform of a kernel function $\phi_{t-d}(t,\tau)$:

$$S_x(t,f)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\phi_{t-f}(t-s,f-\nu)WVD_x(s,\nu)dsd\nu, \tag{12}$$

where:

$$\phi_{t-f}(t,f)=\int_{-\infty}^{\infty}\phi_{t-d}(t,\tau)e^{-j2\pi f\tau}d\tau. \tag{13}$$

For analysis efficiency, the results of implementation of the first approach has been shown. The implemented algorithm consists of two essential steps: computation of an instantaneous smooth autocorrelation function and determination of the Fourier transform of each column of the smooth autocorrelation matrix (the columns represent the time variable). It has been implemented three procedures to carry out the first step. The first one uses 1-dimensional NDRange. In this case, the number of work-items is equal to the number of input samples. The procedure can be carried out with or without using local variables. The second one uses 2-dimensional NDRange. In this case, the number of work-items is equal to the product of the number of input samples and the number of columns of the smooth autocorrelation matrix. The last one uses 3-dimensional NDRange. In this case, the number of work-items is equal to the product of the number of input samples and the number of columns and rows of the smooth autocorrelation matrix. The first kernel needs two "for" loops, the second one "for" loop and the third is carried out without loops, but requires an implementation of procedure for the data exchange between the global and local memory. For implemented the Fourier transform fast radix2, radix8, radix16 and radix32 methods have been used. The results shown in the Fig. 5 and 6 have been carried out using GPU AMD Radeon HD5830 and CPU AMD 4055e. The analyzed signal has consisted of 4096 samples. The Fig. 5 shows the time consumed for determination of a time-frequency representation in dependence of a rank of the function $\phi_{t-d}(t-s,\tau)$ when a rank of smoothing of the function $\phi_{t-d}(t-s,\tau)$ is equal 3. The Fig. 6 shows the time consumed for determination of time-frequency signal representation in dependence of a rank of smoothing of the function $\phi_{t-d}(t-s,\tau)$ when a rank of the function $\phi_{t-d}(t-s,\tau)$ is equal 4095.

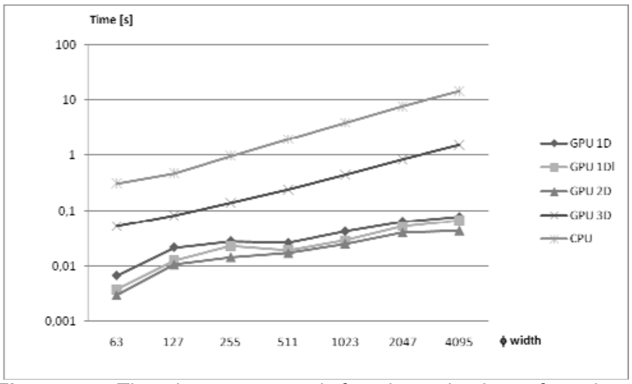

Figure. 5. The time consumed for determination of a time-frequency representation in dependence of a rank of the function $\phi_{t-d}(t-s,\tau)$.

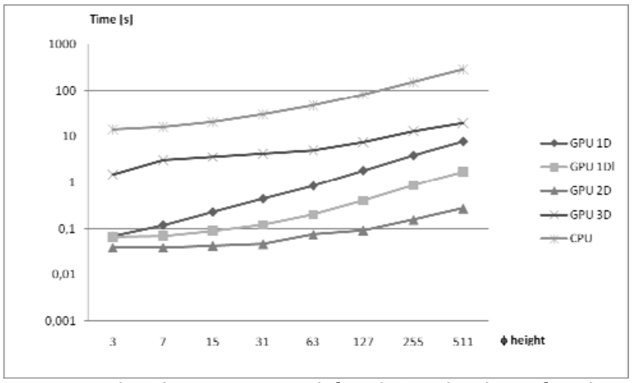

Figure. 6. The time consumed for determination of a time-frequency signal representation in dependence of a rank of smoothing of the function $\phi_{t-d}(t-s,\tau)$.

Conclusions

The paper has addressed the problem of use the GPU for a nonstationary signal analysis. As shown in the paper the proposed approach can reduce the computation time even around 1000x and reduction of 20x can be achieve without any special optimization. Moreover the use of GPUs and OpenGL allows researchers and practitioners to develop numerous practical applications in the area of time-frequency signal analysis.

REFERENCES

[1] Auger F., Hlawatsch F.: "Time-Frequency Analysis", Wiley-ISTE, New York 2008.

[2] Papandreou-Suppappola A.: "Applications in Time-Frequency Signal Processing", CRC Press, New York 2003.

[3] Konopko K.: "A detecton algorithm of LPI radar signals", in Proc. Signal Processing'2007: Algorithms, architecture, arrangement and applications, Poznań, 2007

[4] Grishin Y., Niczyporuk W.: "LFM radar signal detection in the joint time-frequency domain" in Proc. Signal Processing Symposium, Jachranka, 2007.

[5] Khronos Group, The OpenCL Specification Version 1.1, A. Munshi, ed. Khronos Group, 2011.

[6] Kirk D. B., Hwu W. W.: "Programming Massively Parallel Processors: A Hands-on Approach", Morgan Kaufmann, New York 2010.

***Author**: dr inż. Krzysztof Konopko, Politechnika Białostocka, Wydział Elektryczny, ul. Wiejska 45D, 15-351 Białystok, e-mail: krzysiek@teleinfo.pb.edu.pl;*